CCSDS SPACECRAFT UPLINK PROTOCOL IN A SPACE-QUALIFIED ASIC

S. H. Wood, G. S, Bolotin, J. A Donaldson
Jet Propulsion Laboratory
California Institute of 1 ethnology
Pasadena. California

Abstract

1 his paper describes the capabilities of the Hardware Command Decode (1 HCD) application specific integrated circuit (ASIC) developed for the Cassini spacecraft. 1 he ASIC is used on the spacecraft in the Command and Data Subsystem computer to receive and decode uplink commands. 1 he ASIC receives a serial, digital data stream of uplink data from the antenna receiver electronics. Implemented in the ASIC is the "receiving end coding layer" specified in the Consultative Committee for Space Data Systems (C CSDS) 1 elecommand standard. 1 he CCSDS Telecommand standard is a widely used, international standard, The ASIC also supports direct ground control of up to 32 relays and 24 discrete outputs via Virtual Channel O commands. 1 hose relays and discrete outputs allow ground control of critical functions independent of the flight software. 1 he fact that the ASIC also contains a variety of additional features unrelated to uplink that are often needed in spacecraft data systems makes this ASIC particularly useful. The ASIC, produced in Honeywell's RICMOS technology, is radiation hard, single event upset hard, and highly reliable (Class S). The use of space qualified ASICs is an enabling technology for small satellites. ASICs such as the HCD ASIC, which implement part of the CCSDS downlink standard, are expected to have broad applications in small, commercial satellites,

introduction

1 he following topics are addressed in this paper.

(1) Uplink Data Decoding: The uplink data format, data decoding, and error detection and correction functions performed by the ASIC are described in detail,

(2) Critical Enables: 1 he decoding of Virtual Channel O commands by the hardware and control of the Critical Enables is described,

(3) Fault Detection Features: A variety of functions performed by the ASIC that are independent of uplink, but are often needed in spacecraft data systems, are described,

(4) User Interfaces: A overview of the interface to the spacecraft uplink receiver electronics and the host computer is given.

(5) Implementation Description: 1 he ASIC's key performance parameters and the integration of the ASIC into the host subsystem is discussed,

(6) Development Status: 1 he current development status of the ASIC is described.

(7) Related ASICs: A brief description of three other ASICs developed for use in the Cassini Command and Data subsystem is provided. 1 hese three ASICs, along with the HCD ASIC and a processor, perform uplink, downlink, and spacecraft inter-communications.

(1) Uplink Data Decoding

1 he primary function performed by the HCD ASIC is to decode the uplink data stream which is formatted in accordance with the CCSDS Telecommand standard. 1 he ASIC receives serial, digital uplink data from the uplink receiver on the spacecraft. The uplink data stream is composed of three data structures, which are described below.

Acquisition. Sequence: 1 he Acquisition Sequence is an alternating patterns of ones and zeros, starting with either a one or a Zero. It is used at the start of an uplink session to allow the uplink receiver to "lock" onto the data stream. The HCD ASIC requires that this pattern be a minimum of two bytes in length.

Command Link 1 ransmission Unit (Cl 1 U): The CLTU contains the actual data being sent to the spacecraft, The format of the CL TU is discussed below.

Idle Sequence: The Idle Sequence is an alternating pattern of ones and zeros. It allows the uplink receiver to maintain lock in the absence of CLTUs.

A CLTU is composed of three fields, as shown in Figure 1. Each field is described below.

Start Sequence: The Start Sequence consists of 16 bits and has the value EB90 hex if the data polarity is positive, and 146F hex if the data polarity is negative (data polarity is discussed later in this section). The ASIC uses the Start Sequence to identify the beginning of a CLTU.

TeleCommand Transfer Frame(s): The CLTU may contain several TeleCommand (TC) Transfer Frames. Each TC Transfer Frame consists of several TC Codeblocks. Each TC Codeblock consists of 56 information bits, 7 check bits and 1 fill bit, as shown in Figure 2. The check bits are generated using a (63,56) modified Bose-Chaudhuri-Hocquenghem (BCH) code. This code is specified in the CCSDS TeleCommand standard.

Tail Sequence: The Tail Sequence consists of 64 bits and has the value 5555 555555555555 (hex) if the data polarity is positive, and AAAA AA/W AAAA AAAA (hex) if the data polarity is negative. The Tail Sequence is used by the ASIC to identify the end of a CLTU.

Polarity Ambiguity Resolution

Because the uplink data received by the HCD ASIC may be inverted or non-inverted, the ASIC performs polarity ambiguity resolution on the data. When the Start Sequence is 146F hex, the polarity of the data is negative and all subsequent data received will be inverted by the ASIC before being written into software buffers. When the Start Sequence is EB90 hex, the polarity is positive and therefore no inversion is needed,

"Enhanced". Start Sequence

The ASIC also has two user-selectable modes of operation that can be used to determine the start of a CLTU. In one mode, the ASIC uses only the Start Sequence to determine the start of a CLTU. In the second mode, the ASIC uses an "enhanced" start sequence to detect the start of a CLTU. The Enhanced Start Sequence contains an Acquisition Sequence (minimum two bytes long) and then the Start Sequence. Using the Enhanced Start Sequence reduces the probability of the ASIC mistakenly detecting the start of a CLTU in a stream of random data.

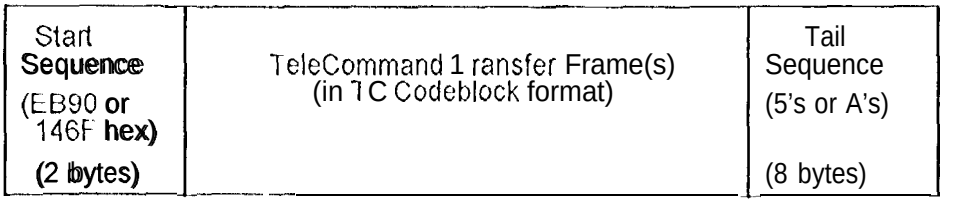


figure 1: Command Link Transmission Unit (CLTU) Format

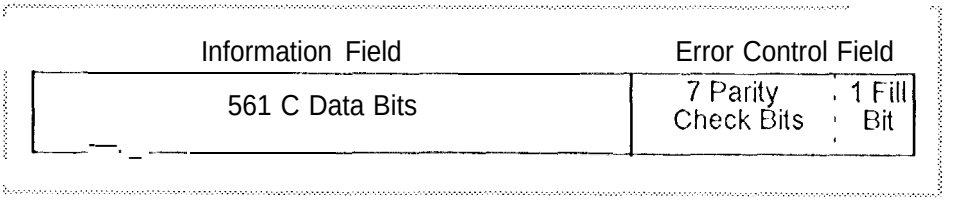


figure 2: Telecommand Codeblock Format

Error Detection and Correction

As described earlier, each Codeblock contains seven parity check bits, which allow error detection and correction to be performed on the Codeblock. The ASIC has two modes of error handling, which is user selectable. In one mode, single errors are corrected, and double bit errors are detected, In the other mode, single, double, and triple bit errors are detected, but there is no correction. If the uplink channel is very noisy, the user may want to have triple bit error detection. If the uplink channel is not very noisy, the user may prefer to have single bit error correction.

TC Transfer Frame Format

The TC Transfer Frame in the CLTU consists of a Frame Header, Frame Data Field, and an optional Checksum, as shown in figure 3. Figure 3 also illustrates how the various levels of data formatting fit together. The format of the Frame Header will be discussed in more detail in the next section.

Software Interface

The HCD ASIC contains two data buffers which are used to pass the uplink data to the software, Each data buffer consists of four 16 bit registers. The bus, each data buffer can hold one 64 bit Codeblock At any point in time, one buffer is the active buffer that the ASIC is writing data into, while software is reading data from the other buffer. The ASIC also contains a status register which allows software to determine which buffers are full, if the buffers have been overrun, if the receiver is in lock, if any errors have been detected in the data, and other status, The ASIC can also be programmed to interrupt the host processor whenever it fills a buffer.

Data Decoding Operation

The state diagram in Figure 4 illustrates the data decoding operation of the ASIC. As will be discussed in the User Interface section, the (CLTU) ASIC receives a LOCK signal from the spacecraft uplink receiver. When the LOCK signal is asserted, it indicates that the input data to the ASIC is valid data. When the LOCK signal is reasserted, it indicates that the data is not valid and so the ASIC ignores the data

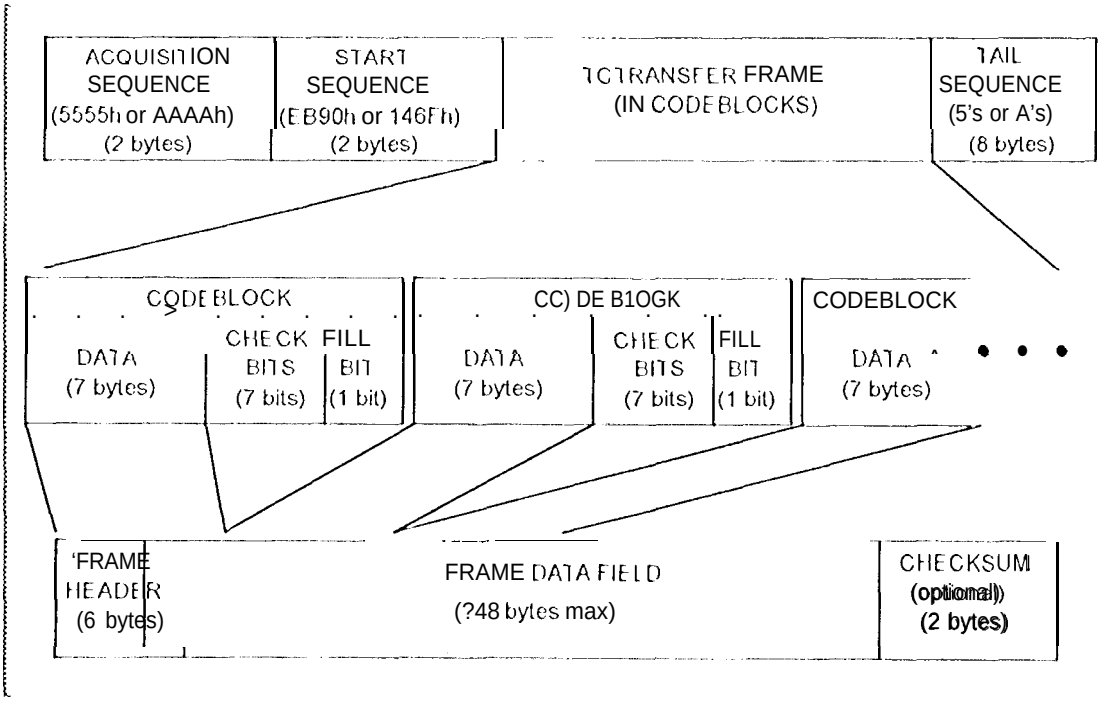


Figure 3: TC Transfer Frame Format

lhe uplink data decode function in the HCD ASIC operates as follows. Whenever the LOCK signal is deasserted, the ASIC transitions to the Inactive State and ignores the uplink data stream. Once the LOCK signal is asserted, the ASIC transitions to the Search State and starts searching through the uplink data stream for the Start Sequence, Once the Start Sequence is detected, the ASIC transitions to the Decode State and starts decoding Codeblocks. When a Codeblock is received, error detection and correction is performed and then Codeblock is placed in a data buffer that can be accessed by software. When a Codeblock with an uncorrectable error is decoded, the ASIC transitions back to the Search State and starts searching for the Start Sequence. Since the 1 ail Sequence contains an uncorrectable error, it forces the ASIC back into the Search State.

It should be noted that except for Virtual Channel O commands (which are discussed in the next section), the format and contents of the data in each Codeblock is a "don't care" to the ASIC. Software must fetch the Codeblock from the data buffers and perform the necessary format checks and data interpretation

(2) Critical Enables

The HCD ASIC allows the ground to directly control up to 3? non-volatile relays and 24 volatile discrete outputs via Virtual Channel O Transfer Frames. These relays and discrete outputs, referred to as Critical Enables, allow the ground to control critical hardware functions independent of flight software.

The virtual channel number is specified in the 1 ransfer Frame F leader. 1 he Frame Header format is shown in Figure 5. 1 he Virtual Channel ID bits are used to logically multiplex a single physical telecommand data channel into 64 logical channels. 1 he ASIC imposes two restrictions on the use of these channels:

(1) Virtual Channel O is reserved for '(hardware commands", which toggle the Critical Enables.
(2) When using Virtual Channel O, the CLTU may contain only one Transfer Frame.

Once the ASIC detects a Start Sequence, it then checks the Virtual Channel ID bits in the Frame Header in the first TC Transfer Frame. When a Virtual Channel O Transfer Frame is detected, the ASIC checks that the Spacecraft ID bits and some of the other bits in the F rame Header are correct. If they are correct, the ASIC then writes to the Critical Enables specified in the Frame Data Field, and places the Codeblock in the software data buffer. When a non-Virtual Channel O Transfer Frame is detected, the ASIC simply places the Codeblock in the software data buffer without checking any of the other bits in the Frame Header

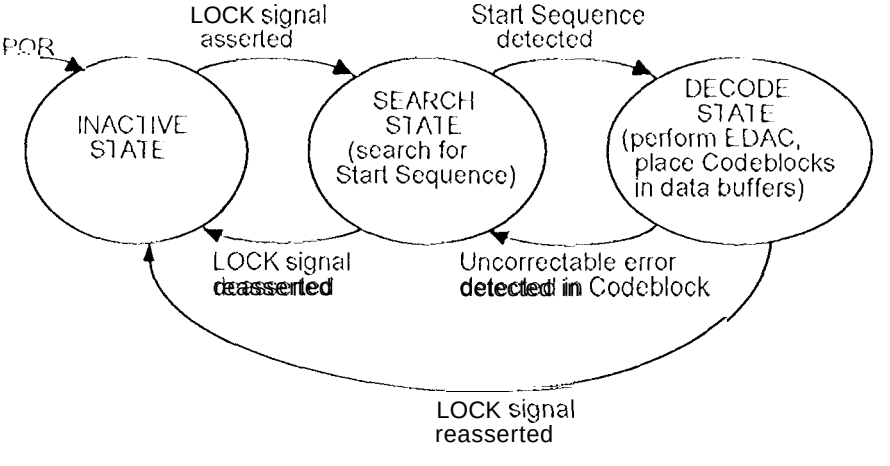


Figure 4: Data Decoding

1 he user can set the spacecraft ID via ten input pins on the ASIC. Anywhere from one to eight Critical Enables can be set with a single Transfer frame. 1 he ASIC contains a mask bit for each Critical E nable. 1 he mask bits are controlled from the ground via Virtual Channel O commands. The ground can allow flight software control of a Critical Enable via the mask bits.

(3) Fault Detection Functions

1 he HCD ASIC provides a variety of fault detection-related functions independent of uplink that are often needed in spacecraft data systems. All of these functions operate independently, and the user has the option of either using or disabling each function. Each function is briefly described below.

(1) A "sophisticated" watchdog timer. Software must write three specific data words to three specific, non-contiguous addresses within a fixed amount of time, or a system reset will be issued.

('2) Interrupt control support: 1 he ASIC has eight input pins that can be connected to generic interrupt sources by the user. 1 he eight inputs are logically ORed together to produce a single interrupt that can be connected to the host processor. 1 he ASIC contains registers which allow software to determine the source of the interrupt, to clear the interrupt, and to individually mask each interrupt source.

(3) Reset control: The reset control function allows up to six different reset sources, three of them user definable, to produce a reset of the host subsystem.

(4) E ight discrete inputs and eight discrete outputs are available which support the exchange of system "health" and other information between redundant systems,

(5) Start-up PROM interface: 1 he ASIC also provides a software interface to start-up PROM. 1 he interface supports up 64 kwords (16 bit words) of PROM. 1 he ASIC was designed to interface with the t larris t IS6617-RH 8kx8 PROM o r equivalent, 1 he ASIC also provides single bit error correction, double bit error detection on the PROM data.

(4) User Interfaces

lwo of the primary interfaces in the IICE) ASIC are the uplink receiver interface, and the Inter-Subassembly Bus (ISB) interface. Each is described below.

Uplink Interface

1 he HCD ASIC interface to the uplink receiver is a serial, synchronous, digital interface. The HCD ASIC receives three signals from the uplink receiver:

(1) 1 he LOCK signal indicates if the uplink receiver is in lock and producing valid data. 1 he ASIC only decodes the uplink data when the LOCK signal is asserted: when the LOCK signal is reasserted, the ASIC ignores the data.

(2) The CLOCK signal is used to clock in the data.

(3) 1 he DATA signal consists of serial, NRZ data. The ASIC can support an uplink data rate of up to 200,000 bits per second.

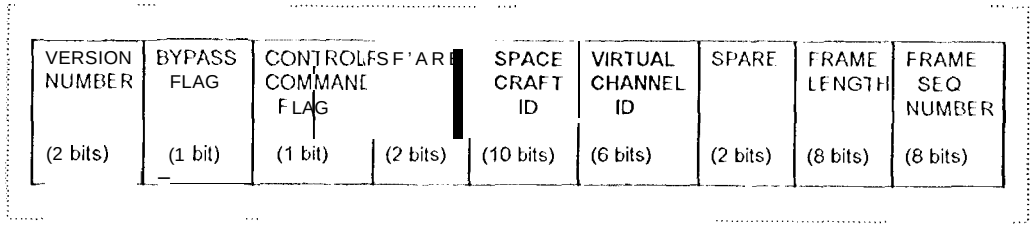


Figure 5: Frame Header Format

1 he uplink interface is illustrated in F igure 6.

Inter-Subassembly Bus (ISB) Interface

1 he HCD ASIC communicates to the host processor via the ISB bus. All of the registers in the ASIC, as well as the start-up PROM, are accessible via the ISB. 1 he ISB is a custom bus developed at JPL. It is a 16-bit, parallel, fully-interlocked asynchronous, multi-n-laster bus, 1 he t ICD ASIC acts only as a slave on the bus. 1 he bus protocol is shown in Figures 7 and 8 below.

A space-qualified field programmable gate array (FPGA) is currently under development at JPl for the Mars Pathfinder spacecraft which will allow the ASIC to interface to a VME bus. 1 he FPGA acts as a translator between the ISB and VME buses.

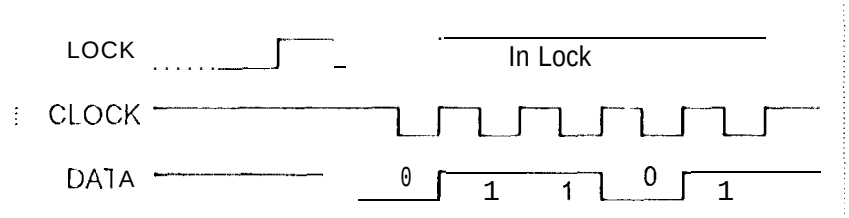
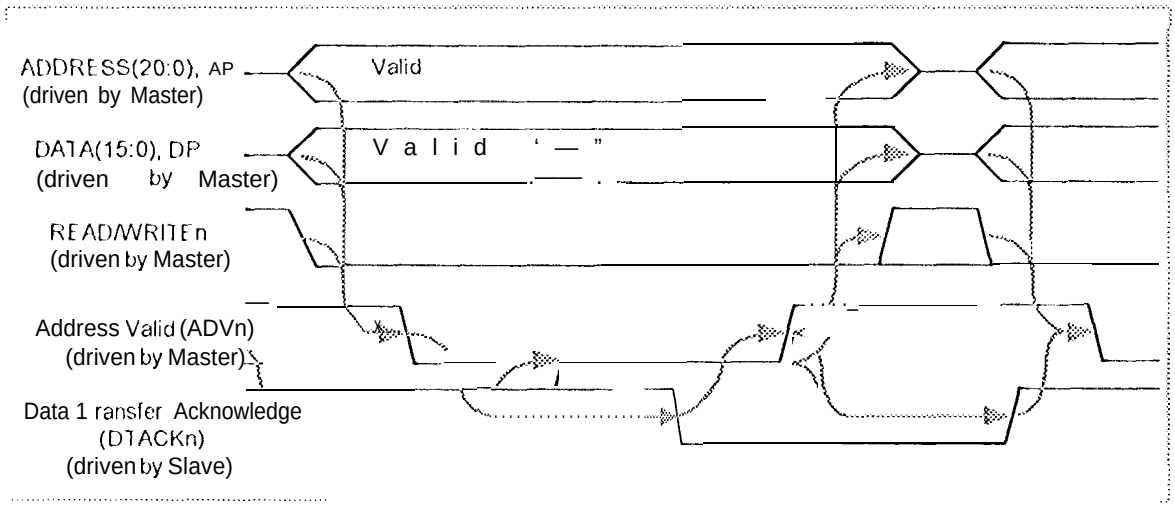


Figure 6: Uplink Interface



F igure 7: ISB Write Cycle

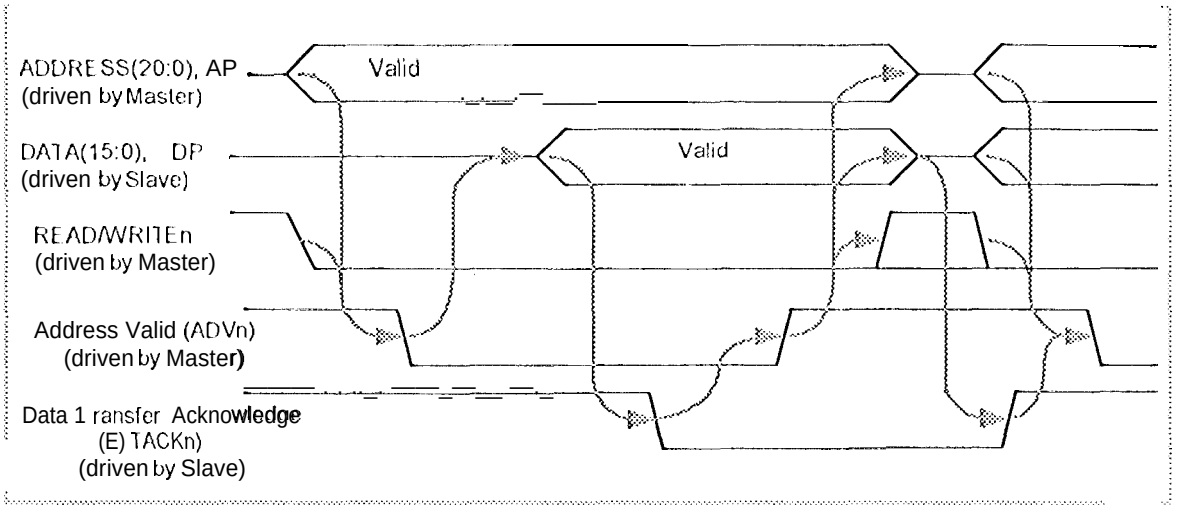


Figure 8: ISB Read Cycle

(5) Implementation Description

The HCD ASIC is implemented in Honeywell's RICMOS, 1.2 micron process. The ASIC is a Class S part produced on Honeywell's QML line, It contains 25,000 gates, Some of its key performance parameters arc shown in 1 able 1.

1 he ASIC requires very few external support chips in order to function, Those support chips that are required are described below:

(1) 1 he Critical Enable relays must be provided external to the ASIC. Also, the ASIC provides only digital outputs with 6 mA drive capability to control the Critical Enable relays. 1 herefore, drive circuits for the relays are also required.

(2) If the start-up PROM capability is used, then the PROM chips must be provided externally, An external decoder chip is also required to generate the chip selects for the PROMs.

(3) Since all of the ASIC outputs have 6 mA drivers, external drivers are typically not required, However, when driving a large capacitive load, external buffering may be required.

(6) Development Status

Initially, a field Programmable Gate Array (F F'GA) version of the ASIC was developed and integrated into the subsystem. Testing of the FPGA design has been successfully completed. 1 he FPGA design has been operating for more than 18 months.

KEY PERFORMANCE PARAMETERS	
Operating Voltage	4,5 to 5,5 volts
Operating Temperature	-55 to + 125 °C
Total Dose Radiation Hardness	>1 Mrad (Si)
Single Event Upset Rate	.0.17 upsets/year (galactic cosmic ray environment)
Dynamic Power Consumption	100 mW (maximum)
Single Event Latch-up	No latch-up @ LET = 120 MeV/mg/cm^2
Clock Frequency	12 MHz
Stuck-At Fault Coverage	>98.5°/0
Package type	256 pin-leaded flat pack

Table 1: Key Performance Parameters

After testing of the FPGA design was completed, a proof-of-design (POD) ASIC was built at Honeywell. The POD ASIC is identical to the ASICs that will be flown in space, except for a reduced level of screening, Testing of the POD ASICs in the subsystem was completed in July, 1994. Fabdication of the flight ASICs is underway at Honeywell. Flight HCD ASICs will be delivered to JPL in December, 1994.

(7) Related ASICs

As shown in Figure 9, besides the HCD ASIC, three other ASICs have been developed for the Cassini spacecraft which implement generic functions required in all spacecraft Command and Data subsystems, The RSDL ASIC supports the CCSDS Packet Telemetry protocol. It performs Reed-Solomon encoding on the telemetry data and formats it into a downlink frame. It also contains a Timing Unit which provides spacecraft time. The SSRIU ASIC provides software with an interface to a Solid State Recorder. The XBA ASIC works in conjunction with the UTMC BCRTM chip to act either as a 1553B RT or BC. The XBA ASIC provides software with a simple interface to the 1553B bus. All of the ASICs communicate with the processor via the ISB bus. The development status of the RSDL, SSRIU, and XBA ASICs is the same as that of the HCD ASIC.

The HCD, RSDL, SSRIU, and XBA ASICs are also being used on the Mars Pathfinder spacecraft, which has a processor with a VME interface. A small FPGA is being developed on that project which provides a translator between the ISB and VME buses

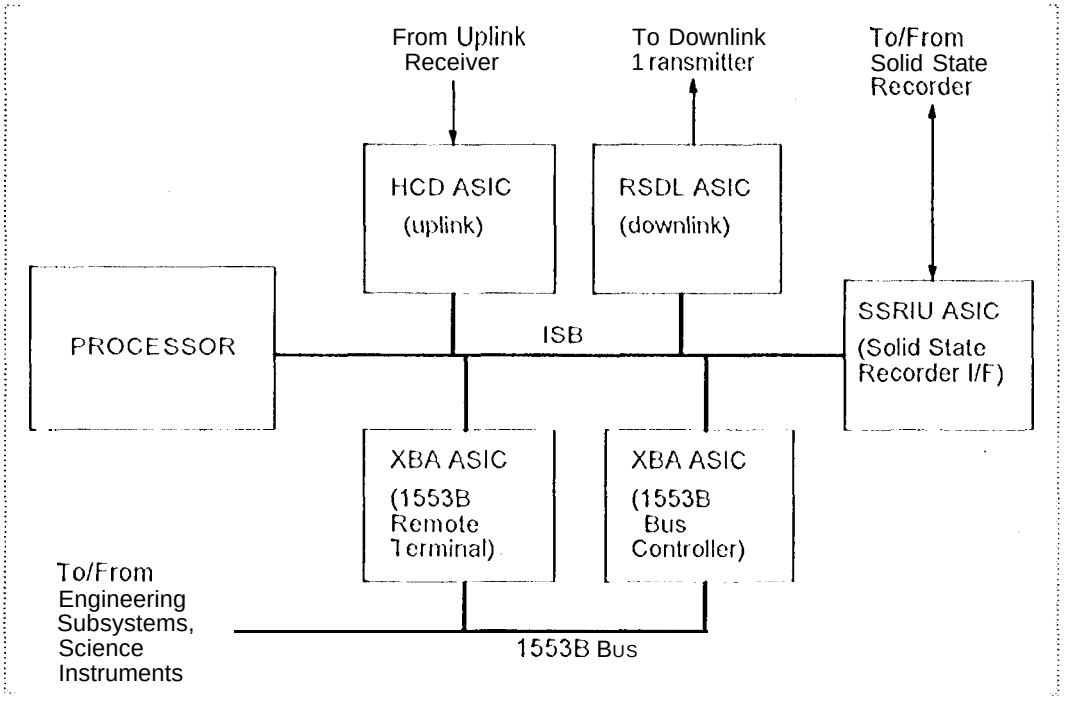


Figure 9: Cassini Command and Data Subsystem Block Diagram

Conclusion

1 he t ICD ASIC provides the user with a valuable link between the digitized uplink data produced by the uplink receiver, and flight software which must process that data. 1 he fact that the ASIC adheres to the CC SDS uplink standard makes it particularly useful. 1 he ASIC's high reliability (Class S) and radiation hardness should meet any user's needs. F unctioning ASICs are currently operating in hardware and software testbeds at JPL and flight ASIC fabrication is underway. The ASICs ISB interface is a generic interface that the user can adapt to operate on a variety of buses. 1 he HCD ASIC, along with three other ASICs developed for the Cassini spacecraft and a processor, can provide the user with a complete Command and Data subsystem. ASIC development is a high cost and high risk task that should be avoided when possible through the use of off-the-shelf ASICs, such as the HCD ASIC.

Acknowledgement

1 he work described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of 1 ethnology, under a contract with the National Aeronautics and Space Administration,

References

(1) Consultive Committee for Space Data Systems (CCSDS), "Telecommand, Part 1, Channel Service", CCSDS 201. 0-B-1, January, 1987.

(?) Consultive Committee for Space Data Systems (CCSDS), "Telecommand, Part 2, Data Routing Service", CCSDS 202. O-B-1, January, 1987.

(3) G. S. Bolotin, "HCD ASIC Functional Specification", Version 2.1, JPL Specification FM516933, 4/1/94.

(4) S. Wood, "Cassini Command and Data Subsystem t lardware/Software Interface Control Document", ES517047, 8/94,

Author's Biographies

James A Donaldson is a member of the technical staff at the Jet F'repulsion Laboratory in Pasadena, CA. He received his BSEE and MSEE degrees from Rensselaer Polytechnic Institute in 1978 and 1979. He currently holds a position of Cognizant Engineer on the Command and Data Subsystem for the Cassini spacecraft. His hobbies include: world history, philosophy, and movies

Steven H. Wood is a member of the technical staff at the Jet Propulsion Laboratory in Pasadena, CA. He received his BSEE and MSEE degrees from UCLA in 1983 and 1988. t Ie currently holds a position of Cognizant Engineer for the Command and Data Subsystem Electronics for the Cassini spacecraft. tie has worked previously on the Galileo, Magellan, and 1 opex spacecraft His hobbies include running and snowboarding.